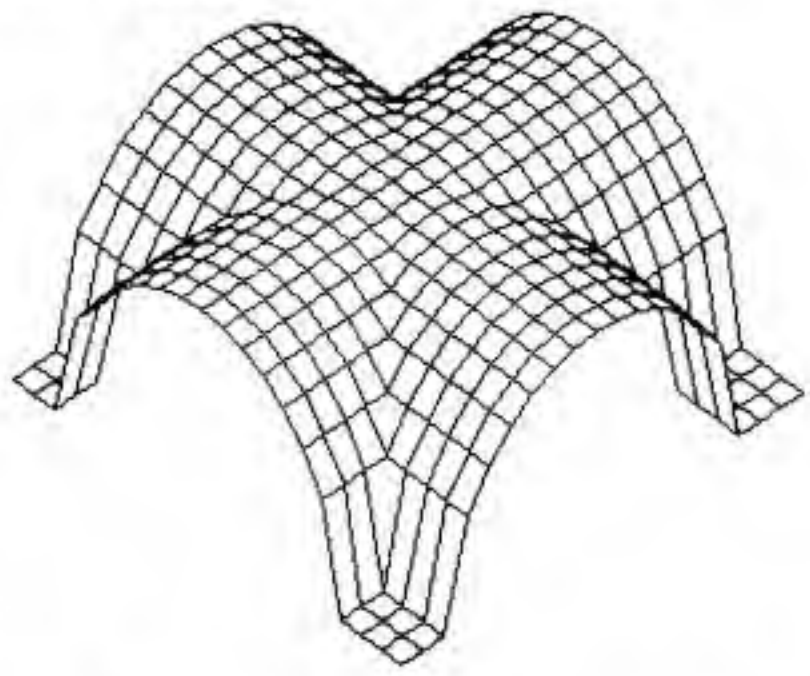
ELBUG

SUPPLEMENT TO BEEBUG

MAY 1985 VOLUME 2 ISSUE 6

3D Surfaces Competition Results

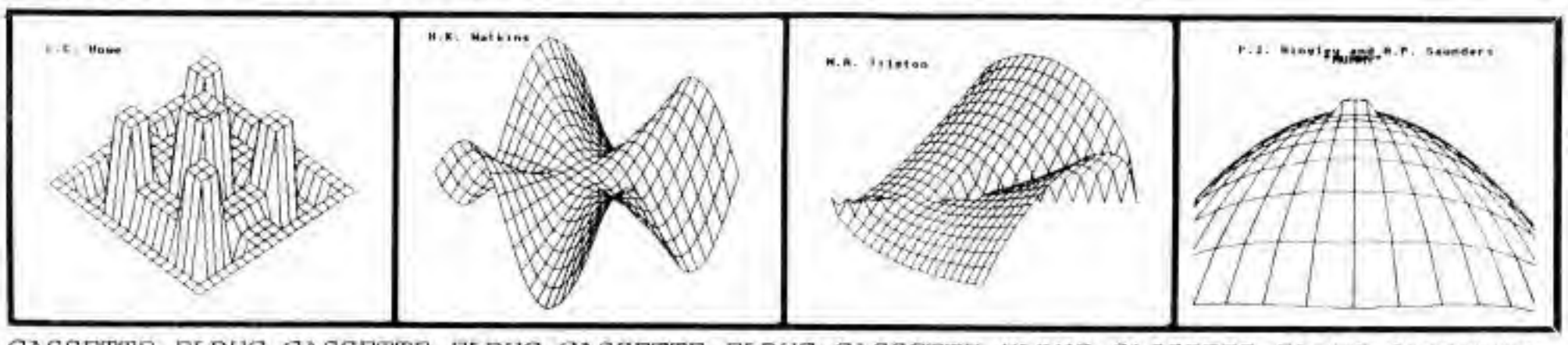
T.J. Fletcher



The competition accompanying the excellent 3D Surfaces program in the Jan/Feb issue of ELBUG was obviously one of the most popular we've ever run. The magazine office was simply knee-deep in entries. Thanks to everyone who entered.

The object of the competition was to provide an equation that would produce the most interesting and pleasing 3D surface. The standard of all the entries was very high but the winner was eventually decided to be Mr T.S.Fletcher of Darlington for his surface depicting two intersecting cylindrical passages. £50 will shortly wing its way to Mr Fletcher.

Several of the other entries were excellent and were only narrowly pipped at the post. We reproduce a few of them here. The range in style is tremendous - from the Castle-like edifice from L.C.Howe (generally considered the runner up) to the more simple but humorous 'Mummy' from P.J.Hingley and A.P.Saunders.



CASSETTE ELBUG CASSETTE ELBUG CASSETTE ELBUG CASSETTE ELBUG CASSETTE ELBUG CASSETTE

The ELBUG Magazine cassette contains a whole host of goodies this month. Apart from the programs from BEEBUG - polar curves, dynamic free memory display, flowchart generator, 'Workshop' sorting examples, the 'Introduction to machine code' examples and the Lunar Bug game - all modified if necessary to give their best on your Electron, there are also the programs from this ELBUG supplement and a special bonus of a full machine code game for the Electron, called 'Shoot the fruit'.

Your Guide to BEEBUG

Just about all the BEEBUG programs and articles are, this month, useful to ELBUG members. There are of course some changes to be made to a few programs and some general comments that can be made on Beeb program conversion.

Conversion of Beeb programs is usually required for one of two reasons - sound and mode 7. As the conversion is usually for the same reasons, it is normally very straightforward.

The Electron has only one sound channel, instead of the BBC micro's three, and no control over the volume of the sound. The Electron is designed so that BBC micro programs making full use of the SOUND command will run on the Electron, though not, of course, producing the full effect. In most cases, then, the Beeb program can be left just as it is with no ill effects.

The Beeb's mode 7 display is of the same format as mode 6 but with a full range of colours and crude graphics. Mode 7 is not available on the Electron, but when the Electron encounters a MODE 7 command it defaults to a mode 6 screen. So, although you should, strictly speaking, change each occurrence of MODE 7 in a BBC program to MODE 6, this isn't really necessary because the Electron treats such commands as MODE 6 anyway. BBC micro programs that include a mode 7 display will therefore run on the Electron but often with a corrupted (and monochrome) display.

The one difference between the Electron and the BBC micro that you cannot alter the programs to account for is the difference in speed. The Electron runs considerably slower than the Beeb in many circumstances. This will affect some programs but not all. Where possible such changes as are possible to reduce the effects of this are given for ELBUG readers.

ISLAND LOGIC MUSIC SYSTEM

The Music System does not operate on the Electron. However, budding Elk-owning musicians will find relief in the second part of our Making Tunes on the Electron in this ELBUG supplement.

POLAR CURVES EXPLAINED

Enter the fascinating world of Polar Curves with this program and add a whole new dimension to Electron graphics.

Equations in polar form can make many interesting graphics shapes quicker and easier to draw on the Electron. Thi article and program will show you just how it's all done.

As an extra bonus we are offering a prize of £50 for the equation that gives the most original and pleasing curve of your own. This competition is of course open to ELBUG members too. See the panel in BEEBUG for the full details.

EXTENDED ASSEMBLER FOR THE 65C02

Alas there is no second processor available for the Electron as yet. There is no way to enable this program to run on the Electron, nor even the unexpanded BBC micro for that matter.

MUSIC SYSTEMS

None of the three music hardware expansion systems reviewed here work on the Electron. The Symphony keyboard relies on the Beeb's sound chip and the others require its expansion ports. Once Acorn gets around to producing the 1MHz bus expansion for the Electron, the excellent Music 500 will then be available to Electron owners.

ULTRACALC 2

Neither Ultracalc nor Ultracalc 2 are available on ROM cartridges for the Electron but if you have one of the several makes of ROM board for your machine you will find that Ultracalc 2 will plug into this and work perfectly.

DYNAMIC FREE MEMORY DISPLAY

This short, but very useful, program will help you to keep tabs on the amount of memory your programs are using, even as they are working.

The program will operate on the Electron exactly as it stands. The program runs a touch slowly on the Electron. This may not worry you but if it does, make the following change:

```
110 D%=25:*FX13 4
```

This alters the update rate of the free memory display from once every fifth of a second to once every half second. This means that the Electron has more time to spend on the actual program being run.

MORE ROMS FOR YOUR BEEB

Unfortunately not more ROMs for your Elk! If you have a ROM board for your Electron heckle your local software house for more Electron compatible ROM software.

FLOWCHART GENERATOR

Keeping track of where your programs are taking you can get pretty difficult at the best of times. A flowchart is one way of mapping out the path of a program. This helpful program will enable you to create clear, tidy flowcharts using your Electron. If you have a Plus 1 or a similar printer interface you can produce a hard copy of your flowcharts too. However, please note that you should not load a machine code printer dump routine at &0D00 (as specified in the text and in line 2430) when using a Plus 1 as the interface uses this area. Try the function key buffer (&0B00) instead.

The flowchart generator makes extensive use of the BBC micro's function keys. Because of the different manner in which the Electron is equipped with function keys, the program will not run as it stands. Make these changes.

Delete lines 120, 130, 140, 1220 and add these lines:

```
1212 IF INKEY-1 THEN A%=A% AND &DF
1213 IF INKEY-2 THEN A%=A%+64
1214 A$=CHR$(A%):A%=INSTR("QWERTYUI
OP",A$)
1215 IF A%<1 OR A%>10 THEN 1260
1216 IF INKEY-2 THEN A%=A%+20
1217 IF INKEY-1 THEN A%=A%+10
1230 XP=GDXP:YP=GDYP
```

These alterations change the control of the program from the function keys to the top line of alphabetic keys: 'QWERTYUIOP'. In other words use the Q key when the text refers to the f0 key, and so on. The Shift and Ctrl keys are used with these alphabetic keys just as the text says they should be used with the function keys. The use of # , ", and the cursor keys remain as in the original program.

ADVENTURE GAMES

The Dungeon Master (curse his mutated mind) has forsaken us Elfin Elk owners. Only Beeb games this month. May the Grand Wizard cast him deep into the dwarf infested forests of Acron for his sins.

WORKSHOP - SEARCHING AND SORTING

Part two of this Workshop series takes a look at searching and sorting of string data. The first program deals with arrays of strings that will all fit into memory at the same time and will work admirably on the Electron. The second program, however, requires a disc drive system to operate as it involves reading in from two files at once - not possible on a cassette machine.

MAKING MUSIC ON THE BEEB (Part 4)

This month Ian Waugh looks at manipulating the multipart tunes created last month. As the Electron can only support a single sound channel, most of these remarks and the programs of last month's article that are extensively referred to, are not applicable to the Electron. However, some of the techniques, such as turning a tune 'upside down' are just as applicable to a solo piece as a three part harmony and will entertain any musical Elk owner.

Part 2 of our own Electron musical tuition series is also here this month and can be found in this ELBUG supplement.

MIXING MODES (Part 2)

This month the workings behind the

Air Raid

by Alan Webster

This month ELBUG presents a special bonus game. Air raid is a very short game that is quick to type in but slow to master. You must help Wally the air raid warden to catch the falling bombs in his bucket of water. Each time Wally successfully catches a bomb the next falls a little faster. If one hits the ground it explodes and Wally cops it.

```
  10 REM PROGRAM AIR RAID
  20 REM AUTHOR  ALAN WEBSTER
  30 REM ELBUG   MAY 1985
  40 REM PROGRAM SUBJECT TO COPYRIGHT
  50 MODE 1:SC%=0:VDU23,1,0;0;0;0;
  60 VDU19,0,4;0;19,3,0;0;23,224,191,1
27,32,17,13,7,3,1,23,225,253,254,4,136,
208,224,192,128,23,226,1,1,1,3,6,12,56,
0,23,227,128,128,128,192,96,48,28,0,23,
228,60,24,60,60,60,60,24,0
  70 X%=19:A$=CHR$32+CHR$224+CHR$225+C
HR$32+CHR$8+CHR$8+CHR$8+CHR$8+CHR$10+CH
R$32+CHR$226+CHR$227+CHR$32
  80 dead=FALSE:R%=10:REPEAT:IFR%<2R%=2
  90 D%=RND(36)+1:O%=0:REPEAT
 100 FORI=1TOR%:X%=X%-(INKEY-67)+(X%>3
5)+(INKEY-98)-(X%<2):VDU17,2:PRINTTAB(X
%-1,30)A$;:NEXT:O%=O%+1
 110 VDU17,3,31,D%,O%,228,31,D%,O%-1,3
2:UNTILO%=29
 120 IFD%=X%ORD%=X%+1PROCsc ELSEPROCbmb
 130 UNTIL dead
 140 FORA=1 TO 4000:NEXT:CLS:*FX21
 150 PRINTTAB(14,15)"Score : ";SC%
 160 REPEAT UNTIL GET=32:RUN
 170 DEFPROCsc
 180 SC%=SC%+10:SOUND 1,-15,1220,2
 190 VDU31,30,0:PRINTSC%;:R%=R%-1
 200 VDU31,D%,O%,32:ENDPROC
 210 DEFPROCbmb
 220 SOUND 0,-15,4,10:dead=TRUE
 230 VDU17,1:PRINTTAB(X%-1,30)A$;
 240 VDU31,D%,O%,32:ENDPROC
```

Mixed Modes program of last month are revealed. Useful reading if you want to find out just why this program cannot work on the Electron.

EXPLORE THE WORLD OF ART AND GRAPHICS

If there is one task that the Electron can handle better than any other micro in its class, that is graphics. These two new books will show you just what is possible with a simple home computer. Some of the results are truly amazing.

BEGINNERS START HERE - INTRODUCING MACHINE CODE (Part 4)

If you're struggling with the complexities of machine code, this Beginners Start Here series is just what you need. This concluding article looks at block moves and simple arithmetic. The programs given will all operate on the Electron without any changes. Purists may like to change the occurrence of MODE 7 to MODE 6. However, the programs poke characters directly to the mode 7 screen. This is not possible in the Electron's mode 6, so recognizable characters will not be displayed, only strange bit patterns will appear. However, the techniques are still amply demonstrated and all ELBUG readers should find this an instructional article.

The reference to page 409 of the Beeb User Guide should be read as page 129 of the Electron manual.

LUNAR BUG

Lunar Bug is a classic computer game that is just ideal for the Electron. Guide the lander to a safe touchdown on the mountainous terrain of the moon.

Lunar Bug works perfectly on the Electron.

HINTS

Many ELBUG readers have asked us to supply details on which of the hints and tips in BEEBUG will work on the Electron, so here goes. These are the hints that are applicable to ELBUG:

BEEPLESS BREAK

USES FOR SQUARE BRACKETS
You cannot press CTRL-[on an Electron (you just get]) so this part will not work.

*SPOOL USES

BUG IN EPSON MX80

BASIC SPEED CONTROVERSY

QUITTING *EXEC

CASSETTE SOFT LABELLING

Making Tunes on the Electron
(Part 2)

Charles Francis continues his quest for musical perfection on the Electron. This month he shows how the single sound channel can be made to perform amazing musical feats.

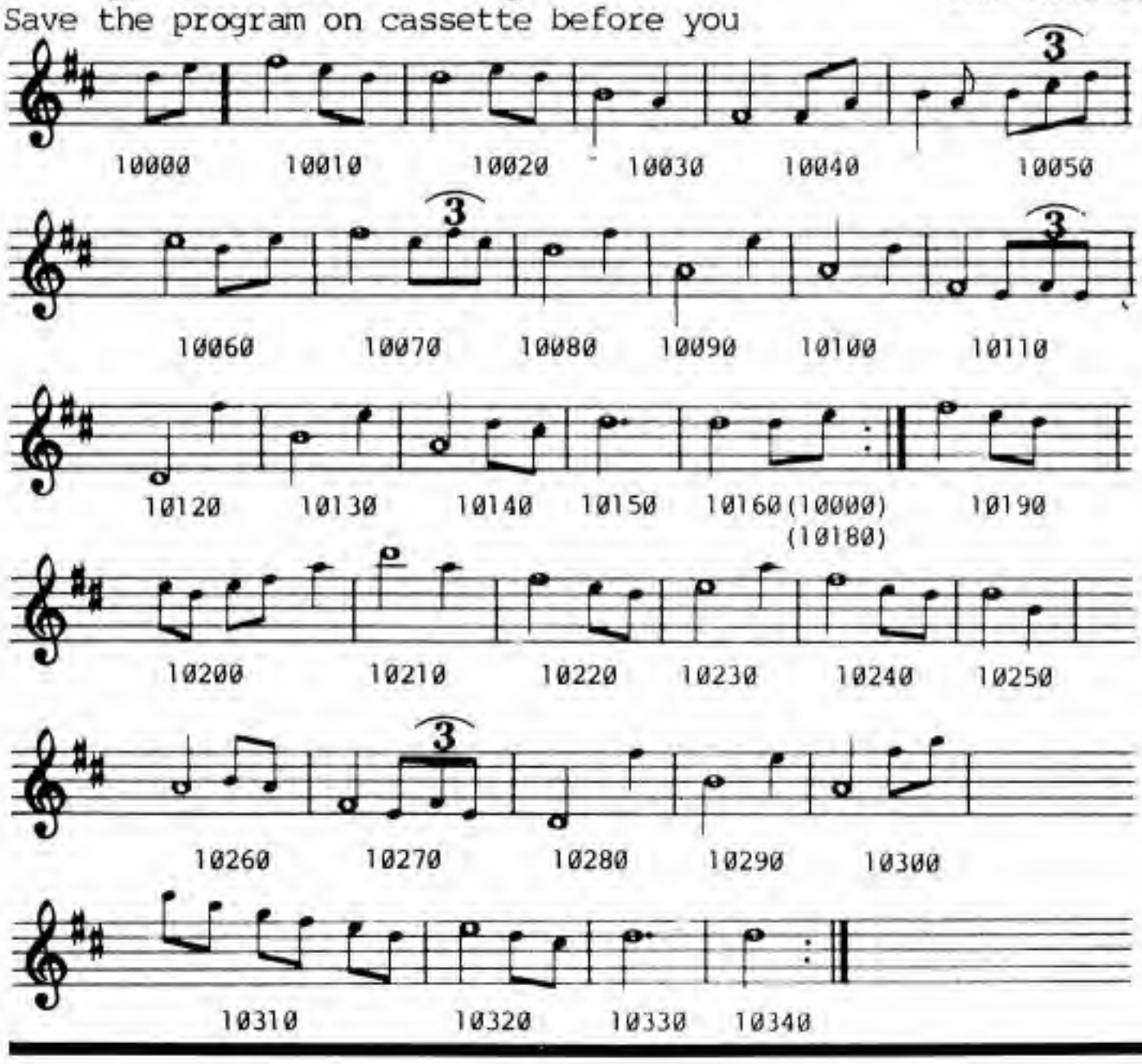
In part 1 (see ELBUG Vol.2 No.3) we saw how to program the Electron to play simple tunes. In this article I will show how a more complex melody can be programmed and ornamented. The Electron can be regarded as a simple melodic instrument, like the penny whistle or the recorder. The embellishment I describe here is derived from that developed for the penny whistle. It is particularly suited to playing traditional folk tunes. The tune I use as my example, 'Si Bheag, Si Mhor' (reputed to be the first written by the brilliant Irish harper Turlough O'Carolan (1670-1738)) shows just how well the Electron can sound, as well as providing a suitable basis for ornamentation.

Enter the program carefully, especially the DATA statements where mis-typing is most likely to occur. Save the program on cassette before you run it.

PROGRAM STRUCTURE

The program itself has a very simple structure, but the data statements need some explaining. Each bar of the tune is written in a single DATA statement, using a separate program line. The number below each bar on the stave refers to the program line in which the data for that bar is stored. Within each data statement the numbers are separated into groups, signifying a single note on the stave, using spaces. Thus in the lead-in bar, in line 10000, there are two notes, each described by six numbers, whereas in the next bar the first note is described by 12 numbers. The reasons for this will become apparent as you read on.

The tune is divided into two parts, one beginning at line 10000, the other at line 10180. The double line at the ends of bars 10170 and bar 10340 on the stave indicate that the preceeding section of music should be played through twice. Then the entire tune is played twice.



NOTE DURATION

PROCplay reads data in groups of three numbers, E%, P%, D%. These are envelope or volume number, pitch value, and duration, as described in the User Guide. Most of the notes include a short rest in their data and so include

a further three data values. Many of the notes that appear single on the stave are programmed as three notes. This is what gives the final result its distinctive flavour. In line 10010, for example, the first 12 data figures all apply to the third note on the stave (three sounded notes and a rest).

The first thing to notice, therefore, is that the duration of the note, shown on the stave, is the sum of the duration of these part notes and rests in the data statements. Thus the first two notes are quavers, for which I have used a total duration of 3, made up of the 2 (the note) and 1 (the rest) in the 3rd and 6th positions. The first note of the next bar is a minim which is four times as long, having a total duration of 12=1+1+10+0 in the 3rd, 6th, 9th, and 12th positions - three notes and a rest.

Notice too the use of triplets in this tune. This is where three notes are played together in the space normally occupied by two quavers, total duration 6. So as not to interrupt the rhythm each note in the triplet must have a duration of 2. There are triplets in bars 10050, 10070, 10110, and 10270. Triplets are introduced to replace pairs of quavers to add variation to the piece.

VIBRATO and TRILL

Originally I planned to do much of the ornamentation using the ENVELOPE command. This turned out to be quite unsatisfactory, partly because ENVELOPE does not synchronize perfectly with SOUND (the imperfection is very slight, but seriously marrs the ornamentation, which needs to be very precise), and partly because it makes the data statements giving the melody much harder to follow, and hence also harder to debug.

The one envelope I have used, in line 120, causes a note to be played slightly vibrato. It is simple to include further ENVELOPES to vary the speed and the emphasis of the vibrato within the piece, but I think there is little to be gained, except in the case of the trill, where the principal note is rapidly alternated with the note immediately above it. If you do use trills you will need two ENVELOPES, since the note above a given note may have a 'P' value either 4 or 8 higher than that note.

TONGING and SLURRING

In part 1, I pointed out the need to put a gap between notes, to give each note a precise starting point. The equivalent technique for the whistle or recorder (which this program imitates) is tonging. If it is desired that the tune flow nicely from one note to the next, as here, the gap length should generally be 0. This is why most notes in the tune are described by a number group ending in 0,0,0, switching the synthesizer off for an instant. For variation and interest some notes are chopped slightly short. In these cases the number groups end in 0,0,1. You should normally only chop notes short either at the end of natural phrases in the tune or at the end of a bar, to add emphasis to the note beginning the next bar. A note to be chopped short is often shown by a dot immediately above or below the note, or by an inverted comma above the score after the note at the end of a phrase, but these notations are often omitted.

In some cases (e.g. the middle notes in bars 10070 and 10110) the sound is not switched off at all in between notes. There is no 0,0,0 or 0,0,1 terminating data. This causes slurring - sliding from one note to the next. Slurring is often shown in a musical score by a curved line connecting the notes to be slurred.

Some experimentation will be necessary to find out how to treat each

note. I would suggest normally you should start off by ending each number group with 0,0,0, then make modifications based on what it sounds like.

SINGLE GRACE NOTES and ROLLS

The main means of ornamentation in this piece is the grace note. This is a short note (duration 1 or 2) sounded at the beginning of the principal note to be played. This has the effect of highlighting that note, so grace notes are often employed at the beginning of a bar, or at a natural peak in the music. They are also used to separate consecutive notes of the same pitch. The note used as a grace note should be a note of the key in which the tune is being played (see Part 1 of this article), and it should normally, though not exclusively, have a pitch value 8 or 12 above that of the principal note. The bar given by line 10020 has two grace notes. If two notes of the same pitch are to be separated using a grace note, and the first one has already been graced from above the second is often graced from below, as in line 10340. If the principal notes are quavers in the same bar this is normally slurred, when it is called a short roll (both principal notes should be the same length here).

Sometimes a bar contains three consecutive quavers of equal pitch. These can be slurred and separated by two grace notes, the first from above (the cut) and the second from below (the tip). This is called a long roll. Alternatively long and short rolls can be used to replace single notes of the same duration.

DOUBLE GRACE NOTES

To add an extra embellishment to key points in the tune a double grace note can be used, either as in line 10010, where the first grace note is of the same pitch as the principal note, or as in line 10160, where the first grace note has a higher pitch than the second, and is used to separate notes of the same pitch, and give extra attack to the beginning of the bar. Triple grace notes can also be used occasionally, to add a flourish, but these are something of an extravagance.

DEBUGGING

It is probably too much to expect to enter the quantity of numerical data required for a program such as this to get a lively version of a tune without a few mistakes. However, if you arrange the data systematically, as described here, then when you run the program you will be able to detect an error in the data by listening to the tune. If rhythm is broken then the 'D' values are wrong in some note; if it sounds out of tune look for an error in the 'P' value. If the tune stops dead at a particular point you are probably missing a comma, so that data for E% is being read into P%.

EXPERIMENTATION

For those who can't play an instrument, but want to learn something about the structure of music, the Electron is an ideal introduction. Start by entering a very simple version of a tune, and then experiment with ornamentation. Not all your experiments will work at first, but by listening to the result you will soon learn much about music, without the need to develop the manual skills normally required of a musician.

```
10 REM Program Si Bheag, Si Mhor
 20 REM Version E0.1
 30 REM Author  Charles Francis
 40 REM ELBUG   May 1985
 50 REM Program subject to copyright
 60 :
100 ON ERROR GOTO 10370
110 :
120 ENVELOPE1,5,1,-2,2,1,1,1,126,0,0,
-126,126,126
130 FOR I%=1 TO 2
```

```
  140 RESTORE
  150 PROCplay
  160 RESTORE
  170 PROCplay
  180 RESTORE 10180
  190 PROCplay
  200 RESTORE 10180
  210 PROCplay
  220 NEXT I%
  230 END
  240 :
 1000 DEFPROCplay
 1010 READ E%,P%,D%
 1020 REPEAT
 1030 SOUND 1,E%,P%,D%
 1040 READ E%,P%,D%
 1050 UNTIL P%<0
 1060 ENDPROC
 1070 :
10000 DATA -15,108,2,0,0,1,  -15,116,2,
0,0,1
10010 DATA 1,124,1,1,136,1,1,124,10,0,0
,0,  -15,116,3,0,0,0,  -15,108,3,0,0,0
10020 DATA -15,116,2,1,108,10,0,0,0,  -
15,124,1,-15,116,2,0,0,0,  -15,108,3,0,
0,0
10030 DATA -15,96,12,0,0,0,  -15,88,6,0
,0,0
10040 DATA -15,76,1,-15,88,1,-15,76,8,0
,0,2,  -15,76,2,0,0,1,  -15,88,2,0,0,1
10050 DATA -15,96,1,-15,104,1,-15,96,6,
0,0,1,  -15,88,2,0,0,1,  -15,96,1,0,0,1
,  -15,104,1,0,0,1,  -15,108,1,0,0,1
10060 DATA -15,116,1,-15,124,1,-15,116,
9,0,0,1,  -15,108,2,0,0,1,  -15,116,2,0
,0,1
10070 DATA -15,124,1,1,136,1,1,124,10,0
,0,0,  -15,116,2,  -15,124,2,  -15,116,
1,0,0,1
10080 DATA -15,108,11,0,0,1,  -15,124,6
,0,0,0
10090 DATA -15,104,1,-15,96,10,0,0,1,
-15,116,6,0,0,0
10100 DATA -15,96,1,-15,88,10,0,0,1,  -
15,108,6,0,0,0
10110 DATA -15,76,1,-15,80,1,1,76,9,0,0
,1,  -15,68,2,  -15,76,2,  -15,68,1,0,0
,1
10120 DATA 1,60,11,0,0,1,  -15,128,1,-1
5,124,5,0,0,0
10130 DATA -15,96,11,0,0,1,  -15,124,1,
-15,116,5,0,0,0
10140 DATA -15,88,11,0,0,1,  -15,116,1,
-15,108,1,0,0,1,  -15,104,2,0,0,1
10150 DATA 1,108,17,0,0,1
10160 DATA -15,124,1,-15,116,1,-15,108,
8,0,0,2
10170 DATA -1,-1,-1
10180 DATA -15,108,2,0,0,1,  -15,116,2,
0,0,1
10190 DATA 1,124,1,1,136,1,1,124,10,0,0
,0,  -15,116,3,0,0,0,  -15,108,3,0,0,0
10200 DATA -15,124,1,-15,116,2,0,0,0,
-15,108,3,0,0,0,  -15,116,3,0,0,0,  -15
,124,3,0,0,0,
-15,144,1,-15,136,4,0,0,1
10210 DATA 1,152,2,1,144,10,0,0,0,  1,1
36,6,0,0,0
10220 DATA 1,124,11,0,0,1,  -15,116,3,0
,0,0,  -15,108,3,0,0,0
10230 DATA -15,116,11,0,0,1,  -15,136,6
,0,0,0
10240 DATA 1,124,11,0,0,1,  -15,116,3,0
,0,0,  -15,108,3,0,0,0
10250 DATA -15,116,1,-15,108,10,0,0,0,1,
   -15,96,6,0,0,0
10260 DATA 1,88,11,0,0,1,  -15,104,1,-1
5,96,2,0,0,0,  -15,88,3,0,0,0
10270 DATA 1,68,1,1,76,10,0,0,0,1,  -15,6
8,2,0,0,0,  -15,76,2,0,0,0,  -15,68,1,0
,0,1
10280 DATA 1,60,11,0,0,1,  -15,124,6,0,
0,0
10290 DATA 1,96,1,1,104,1,1,96,9,0,0,1,
  -15,116,6,0,0,0
10300 DATA 1,88,1,1,96,1,1,88,9,0,0,1,
 -15,124,2,0,0,1,  -15,136,2,0,0,1
10310 DATA -15,152,1,-15,144,1,0,0,1,
-15,136,2,0,0,1,  -15,128,2,0,0,1,  -15
,124,2,0,0,1,  -15,116,2,0,0,1,  -15,10
8,2,0,0,1
10320 DATA 1,116,1,1,124,1,1,116,9,0,0,
1,  -15,116,2,0,0,1,  -15,108,2,0,0,1
10330 DATA 1,128,1,1,124,1,1,116,1,1,10
8,15,0,0,0
10340 DATA -15,104,2,-15,108,9,0,0,1
10350 DATA -1,-1,-1
10360 :
10370 ON ERROR OFF
10380 MODE 6
10390 IF ERR<>17 THEN REPORT:PRINT " at
 line "ERL
10400 END
```